

Detecção de Potenciais Focos do *Aedes aegypti* em Vídeos Aéreos Usando Redes Neurais

Wesley L. Passos¹, Eduardo A. B. da Silva¹, Sergio L. Netto¹, Jonathan B. Martins², Yan B. Costa², Gabriel M. Araujo², Amaro A. de Lima².

Resumo—O mosquito *Aedes aegypti* é o vetor de doenças como dengue, zika, chikungunya e a febre amarela urbana, as quais, todos os anos, afetam milhões de pessoas. A principal forma de combate a essas doenças ainda é a busca e eliminação dos criadouros do mosquito. Nesse trabalho, apresentamos uma ampla base de dados de vídeos aéreos obtidos com um drone, onde os objetos de interesse são demarcados por suas respectivas *bounding boxes*, e contribuimos com um sistema de detecção baseado em redes neurais profundas de possíveis focos do *Aedes aegypti*. Para rastrear os objetos ao longo dos quadros, é feito um alinhamento espacial utilizando *phase correlation*. Com os quadros alinhados, é possível registrar os objetos detectados, minimizando falsos positivos e corrigindo falsos negativos. Utilizando a ResNet-101-C4 como *backbone*, é possível obter um *F1-score* de 0,74 na detecção de objetos da classe pneus ao longo da base de dados proposta.

Palavras-Chave—*Aedes aegypti*, visão computacional, processamento de imagem, detecção de objetos.

Abstract—The *Aedes aegypti* mosquito is the transmitter of diseases like dengue, zika, chikungunya, and urban yellow fever, which affect millions of people every year. The main control for these diseases is done by searching and eliminating the mosquitoes breeding grounds. In this work we introduce a comprehensive database of aerial videos recorded with a drone, where all objects of interest are identified by their respective bounding boxes, and describe an object detection system based on deep neural networks. We employ phase correlation to obtain the spatial alignment between the video frames and track the objects. By doing so, we are capable of registering the detected objects, minimizing false positives and correcting most false negatives. Using the ResNet-101-C4 as backbone, it is possible to obtain 0.74 in terms of *F1-score* on the proposed dataset.

Keywords—*Aedes aegypti*, computer vision, image processing, object detection.

I. INTRODUÇÃO

O mosquito *Aedes aegypti* é o principal transmissor de doenças causadas por arbovírus (termo que designa o vírus transmitido por artrópodes), como dengue, zika, chikungunya e febre amarela urbana [1]. Uma estimativa diz que cerca de 390 milhões de pessoas são infectadas por ano no mundo [2]. Em 2019, somente as Américas notificaram mais de 2,7 milhões de casos de dengue; mais de 2 milhões apenas no Brasil [3]. A infecção pelo vírus zika durante a gravidez está correlacionada à microcefalia e outras deformações congênitas nos bebês; crianças nessas condições raramente se desenvolvem normalmente [4]. Chikungunya pode desencadear problemas neurológicos e cardíacos. Pode ainda incapacitar os infectados

por longos períodos e levar à morte de idosos [5]. A febre amarela causou 483 mortes no Brasil entre julho de 2017 e junho de 2018, um aumento significativo quando comparado a 262 mortes no mesmo período de 2016/2017 [6]. Estas doenças têm um grande impacto na economia. Uma pesquisa realizada em 17 países das Américas Latina e Central estima que o custo das epidemias de dengue nesses países exceda US\$ 3 bilhões por ano, sendo US\$ 1,4 bilhão apenas no Brasil [2]. Esses fatos tornam o arbovírus transmitido pelo *Aedes aegypti* um dos principais problemas de saúde global.

Exceto no caso da febre amarela, não há vacina nem fármacos antivirais específicos para as doenças citadas. Assim, a melhor forma atual de combatê-las é através do controle e eliminação de possíveis focos de mosquitos [7]. O *Aedes aegypti* se reproduz em água limpa e estagnada, portanto, qualquer recipiente que permita o armazenamento de água (caixas d'água, pneus, baldes, vasos de plantas e outros) são potenciais criadouros. Estes objetos estão em toda parte, o que pode tornar o monitoramento e o controle do mosquito, na falta de suporte técnico adequado, caro, demorado e ineficiente.

Atualmente, *drones* já são usados pelas organizações para inspecionar locais de difícil acesso. No entanto, os vídeos adquiridos costumam ser examinados por um especialista que identifica através de inspeção visual os pontos de reprodução, o que tende a ser demorado, cansativo e, conseqüentemente, propenso a falhas.

Assim, nesse trabalho, propomos um sistema para automatizar o processo de análise das imagens, aplicando técnicas de aprendizado de máquina e visão computacional, ajudando o especialista na localização de focos relevantes de mosquitos. Para tal, apresentamos uma extensão da base de dados em [8], com mais vídeos (incluindo cenários reais), objetos e variação maior de altitude do *drone*. Para ilustrar o conceito, pneus são utilizados como objeto de interesse. A detecção dos pneus é feita utilizando a arquitetura *Faster R-CNN* em todos os quadros do vídeo. Em razão das características da base de dados utilizada, podemos assumir que o movimento entre os quadros é puramente translacional. A Técnica de *Phase Correlation* é utilizada para medir esse deslocamento e efetuar a compensação espacial entre os quadros em uma janela temporal. Essa compensação permite adotar uma heurística para efetuar o registro entre as detecções, eliminando boa parte dos falsos positivos e corrigindo a ampla maioria dos falsos negativos.

O artigo está organizado da seguinte forma: A Seção II contém uma revisão de literatura, descrevendo alguns trabalhos relacionados na área. O método proposto está descrito na

¹PEE/COPPE/DEL/POLI, Universidade Federal do Rio de Janeiro, Cx. P. 68504, Rio de Janeiro, RJ, 21945-970, Brasil. ²Centro Federal de Educação Tecnológica, Nova Iguaçu, RJ, 26041-271, Brasil. E-mails: {wesley.passos, sergioln, eduardo}@smt.ufrj.br. {jonathan.martins,yan.costa}@aluno.cefet-rj.br. {gabriel.araujo, amaro.lima}@cefet-rj.br.

Seção III, incluindo uma descrição da base de dados utilizada. A Seção IV contém resultados obtidos e uma discussão acerca desses resultados. A conclusão está na Seção V.

II. TRABALHOS RELACIONADOS

Os autores em [9] propõem um sistema para identificar possíveis criadouros de mosquitos em imagens com georreferenciamento enviadas pela população. A qualidade destas imagens é avaliada a fim de rejeitar imagens com altos níveis de distorções ou artefatos. As imagens são convertidas em vetores de atributos usando o modelo de *bag of visual words* por meio do descritor SIFT (do inglês, *scale-invariant transform*) para treinar um classificador do tipo SVM (do inglês, *Support Vector Machine*). De acordo com a classificação, o sistema produz um mapa de calor destacando as regiões com maior risco de proliferação de mosquitos. Os autores de [10] usam um classificador SVM treinado para detectar poças de água em trechos de vídeos capturados por um quadricóptero. O trabalho em [11] treina um conjunto de classificadores do tipo *naive Bayes* a partir de vetores de atributos calculados utilizando o descritor SURF (do inglês, *speeded-up robust features*) em imagens térmicas e de nível de cinza para detectar água parada. Em [12], um classificador *Random Forest* é treinado usando atributos extraídos do espaço de cores HSV para detectar pneus e água parada.

Os autores de [13] usam imagens do *Google Street View*, de pesquisa de imagens do Google e do conjunto de dados *Common Objects in Context* (COCO) para treinar e testar uma rede neural *Faster R-CNN*, a fim de detectar os seguintes objetos: pneu, balde, lixeira, copo, vaso de planta, vaso, tigela e jarro. As detecções são utilizadas para compor um painel mostrando áreas de risco. Em um conjunto de teste, o sistema descrito em [13] obteve uma acurácia até de 0.91 (*F-score*).

Os resultados reportados em [13] são bastante promissores, mas, devido a particularidades das bases de dados, sua aplicabilidade se restringe a vias públicas. As imagens aéreas possuem um bom potencial de mostrar grandes focos de mosquito em áreas privadas e/ou de difícil acesso. Os artigos em [8], [14] propuseram bases de dados mais adequadas a esse tipo de tarefa. Em [14] é possível encontrar um método que utiliza histogramas nos canais HSV e fluxo óptico para extrair atributos e *Random Forest* para fazer classificação de pneus e água parada. Os resultados são promissores, mas a base de dados proposta é pequena e possui pouca variabilidade.

A base de dados proposta em [8] possui uma variabilidade maior, mas isso ainda é um problema relevante quando se quer utilizar técnicas de aprendizado profundo. Em [15] é possível encontrar um método de *data augmentation* que permite aumentar o conjunto de objetos de interesse nessa base de dados.

Em [16], é descrito um método baseado em *Faster R-CNN* para detectar pneus na base de dados [8]. Apesar da pouca variabilidade da base de dados, os resultados são encorajadores.

Neste trabalho, apresentamos nossa base de dados de vídeos aéreos, em desenvolvimento e crescimento ininterruptos desde 2018. Trata-se de uma extensão da base de dados em [8], com a adição de mais vídeos e objetos de interesse. Ela

possui vários vídeos de diferentes locais, altitudes e disposição de objetos. Os vídeos são anotados densamente (quadro a quadro), o que é de grande valia no desenvolvimento de um detector automático para os objetos de interesse. Também empregamos um detector de objetos baseado na *Faster R-CNN* para realizar a detecção dos objetos. Além da extensão da base de dados, uma importante contribuição deste trabalho está na forma como os objetos detectados são rastreados para eliminar inconsistências na detecção. A técnica de *Phase Correlation* é aplicada em quadros sucessivos para alinhá-los espacialmente em uma dada janela de tempo. Em seguida, uma heurística é empregada para: i) associar detecções diferentes de um mesmo objeto; ii) eliminar falsas detecções (falsos positivos) e; iii) corrigir as detecções faltantes (falsos negativos).

III. MÉTODO PROPOSTO

Nesta seção, apresentamos as etapas para realizar a detecção de possíveis focos do *Aedes aegypti*. Primeiro, apresentamos nossa base de dados, que contém um conjunto de dados de vídeo representativo, adquirido usando um *drone*, com objetos marcados considerados como potenciais locais de reprodução de mosquitos. Em seguida, falamos brevemente sobre o detector de objetos usado para detectar tais possíveis criadouros de mosquitos. Por fim, mostramos a técnica para rastreamento dos objetos.

A. Base de dados

O projeto da base de vídeos desenvolvida no âmbito deste trabalho está descrito em [8]. Essa base contém diversos tipos de objetos que são potenciais criadouros de mosquito, anotados quadro a quadro, com caixas delimitadoras, de forma a permitir uma abordagem supervisionada. Ela possui também a telemetria do *drone*, permitindo assim a localização geográfica de cada objeto de interesse. A câmera do *drone* foi calibrada usando o método de Zhang [17] para minimizar as distorções da lente.

A base de dados utilizada neste artigo é composta por 18 sequências de vídeo em 9 locações diferentes. Os vídeos foram adquiridos em dias diversos e, conseqüentemente, com diferentes condições de iluminação. Três dessas locações são reais. Nas demais os objetos de interesse foram inseridos manualmente na cena em posições aleatórias antes da gravação dos vídeos. Todos os vídeos foram gravados com resolução 4K a 30 Hz, com a duração variando de 23 s a 327 s. As altitudes do *drone* também variaram, sendo que 4 vídeos foram obtidos com altitude de 10 m, 3 com 15 m, 1 com 16 m, 4 com 20 m e 6 com altitude de 40 m. Diversos objetos foram manualmente anotados: garrafas, baldes, piscinas, poças, pneus e caixas d'água. A Figura 1 apresenta alguns exemplos de quadros da base de dados.

B. Detector de objetos

A detecção de objetos consiste na localização e classificação de uma ou várias instâncias de objetos em imagens ou vídeos. O detector atribui a cada imagem um conjunto de caixas delimitadoras, contendo suas coordenadas e rótulos.



Fig. 1: Exemplos de quadros da base de dados. Os objetos de interesse podem ser identificados com um zoom dos quadros.

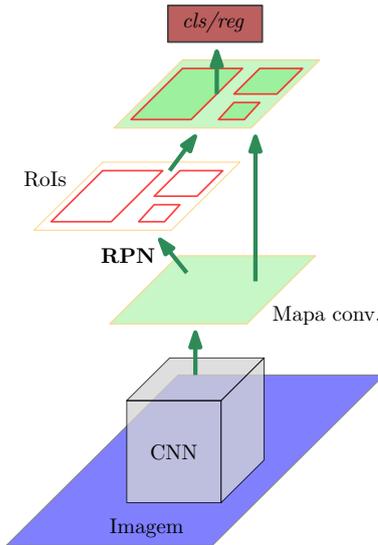


Fig. 2: *Faster R-CNN*.

As redes neurais convolucionais (CNN) têm demonstrado bom desempenho na detecção de objetos [18]. A *Faster region-based convolution neural network (Faster R-CNN)* [19] é uma meta-arquitetura baseada em CNNs que vem apresentando excelentes resultados em várias aplicações e competições [18]. A *Faster R-CNN*, ilustrada na Figura 2, é composta basicamente por um extrator de atributos, uma rede geradora de propostas (RPN, do inglês *region proposal network*) e um módulo de classificação e regressão (cls/reg). O extrator de atributos, que geralmente é uma CNN profunda, extrai os mapas de atributos convolucionais das imagens de entrada. Com base nesses mapas, a RPN gera potenciais regiões de interesse (RoI), que são entregues ao módulo de classificação e regressão (camadas densamente conectadas) que identificam as classes das RoIs e as refinam, prevenindo um deslocamento para as caixas delimitadoras.

C. Rastreamento dos objetos

Devido às características da base de dados, é possível assumir que o movimento entre quadros adjacentes é puramente translacional. Técnicas como a *phase correlation* [20] permitem efetuar de maneira computacionalmente eficiente o registro entre duas imagens sem rotações ou mudanças de escala, como é o caso dos vídeos da nossa base. O deslocamento translacional dos objetos detectados entre quadros sucessivos pode ser obtido como descrito a seguir. Considere que um quadro $\mathbf{f}_{n+1}(x,y)$ é uma versão transladada do quadro $\mathbf{f}_n(x,y)$,

com deslocamento $\mathbf{d} = (x_0, y_0)$. Logo:

$$\mathbf{f}_{n+1}(x,y) = \mathbf{f}_n(x - x_0, y - y_0). \quad (1)$$

De acordo com a propriedade de translação de Fourier, as transformadas de \mathbf{f}_n e \mathbf{f}_{n-1} estão relacionadas da seguinte forma

$$\mathbf{F}_{n+1}(\xi,\eta) = e^{-j2\pi(\xi x_0 + \eta y_0)} \mathbf{F}_n(\xi,\eta). \quad (2)$$

Assim sendo, o espectro de potência cruzado normalizado entre as duas imagens é dado por

$$\frac{\mathbf{F}_{n+1}(\xi,\eta)\mathbf{F}_n^*(\xi,\eta)}{|\mathbf{F}_{n+1}(\xi,\eta)\mathbf{F}_n(\xi,\eta)|} = e^{-j2\pi(\xi x_0 + \eta y_0)}, \quad (3)$$

onde \mathbf{F}^* é o complexo conjugado de \mathbf{F} . Idealmente, a transformada inversa de Fourier da Equação (3) é o impulso $\delta(x_0, y_0)$. Em outras palavras, esta transformada inversa tenderá a ter todos os seus elementos aproximadamente zero, exceto na posição que corresponde ao deslocamento desejado.

Neste trabalho a técnica de *phase correlation* é utilizada para obter o deslocamento espacial dos quadros em uma janela temporal de t quadros com referência no quadro central n . Esse deslocamento é aplicado para alinhar espacialmente cada um desses quadros. Detecções em quadros sucessivos alinhados são consideradas do mesmo objeto quando o IoU $\geq 0,5$ [16] entre os *bounding boxes*. O objetivo do alinhamento é observar o comportamento dos objetos detectados pelo modelo ao longo dos quadros e eliminar os resultados inconsistentes (falsos positivos e falsos negativos). Para ilustrar esse procedimento, considere o exemplo com janela temporal de $t = 3$ quadros na Figura 3. Os retângulos menores em linha cheia representam as detecções (saídas do modelo) nos $t = 3$ quadros consecutivos ($n - 1, n, n + 1$). O retângulo tracejado azul representa um objeto que o modelo falhou em detectar. O retângulo vermelho indica um falso positivo. Cada retângulo contém um número, que representa um objeto específico. Os retângulos com o mesmo número são detecções do mesmo objeto em um quadro diferente (*bounding boxes* com IoU $\geq 0,5$ entre quadros alinhados). Assim, o objeto 2 representa um falso negativo, enquanto o objeto 3 representa um falso positivo no quadro n . As saídas do modelo são utilizadas para compor a seguinte matriz:

		objeto			
		0	1	2	3
quadro	$n - 1$	1	1	1	0
	n	1	1	0	1
	$n + 1$	1	1	1	0
----- decisão:		1	1	1	0

(4)

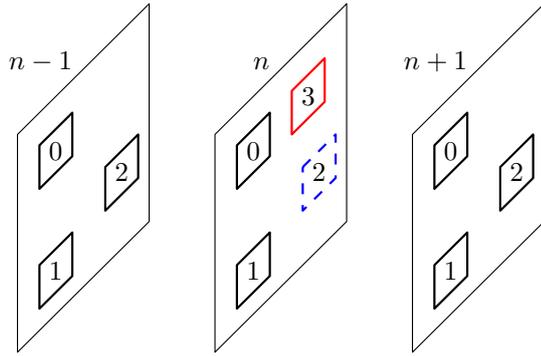


Fig. 3: Os retângulos em linha cheia indicam detecções em três quadros sucessivos, enquanto os retângulo em linha tracejada indica uma detecção que deveria ter ocorrido, mas não ocorreu (falso negativo). O retângulo vermelho é um falso positivo.

onde os valores binários na submatriz em colchetes na parte de cima da linha tracejada indicam a presença (ou não) de um objeto específico em um quadro. Uma votação é aplicada nas colunas dessa submatriz para compor uma decisão (expressão abaixo da linha tracejada da Equação (4)) do que finalmente fará parte do quadro n . Assim, se v é o percentual dos quadros em uma janela em que um objeto está presente, a detecção de um objeto será mantida se $v \geq 50\%$. Com esse procedimento, na Figura 3 o objeto 2 (terceiro elemento do vetor de decisão) foi incluído, enquanto o objeto 3 (último elemento) foi removido do quadro n .

IV. RESULTADOS E DISCUSSÕES

Uma das métricas utilizadas para avaliar as detecções é o *average precision* com $\text{IoU} \geq 0,5$ (AP_{50}) [16], bastante utilizada no contexto de detecção de objetos. Porém, esta métrica sozinha não fornece a ideia da quantidade de verdadeiros positivos (TP), falsos positivos (FP) e falsos negativos (FN), que também serão analisados. Note que verdadeiros negativos (TN) não são aplicados neste contexto. Para agregar as informações de TP, FP e FN, são reportados também os valores de *precisão* (Pr), *revocação* (Rc) e *F1-score* (F_1), dados, respectivamente, por:

$$\text{Pr} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad \text{Rc} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad \text{e} \quad (5)$$

$$F_1 = 2 \cdot \frac{\text{Pr} \cdot \text{Rc}}{\text{Pr} + \text{Rc}}. \quad (6)$$

A. Detecção dos objetos utilizando Faster R-CNN

No total, foram treinados 4 modelos diferentes de Faster R-CNN, utilizando ResNets, com 50 ou 101 camadas, como extrator de atributos (*backbone*). Para alimentar a RPN e o módulo de regressão e classificação, foram extraídos os mapas convolucionais da última camada convolucional do 4º estágio (C4) [21], e alternativamente foi utilizado o esquema de *feature pyramid networks* (FPN) [22].

Vídeos de duas locações diferentes foram separados para a fase de teste e os demais usados no treinamento. Os vídeos são amostrados a cada 30 quadros. Durante o treinamento são considerados positivos os casos de RoIs que possuem

a IoU com as anotações maior que 0,5, sendo todos os demais casos considerados negativos. Todas as imagens são redimensionadas de forma que a borda mais curta não ultrapasse a resolução de 800 *pixels*, mantendo a razão de aspecto. Utilizamos um *batch* de 4 imagens, amostrando 32 RoIs (limitados pela memória da GPU) com proporção 1:1 entre positivos e negativos. Os modelos são inicializados com os pesos da base de dados ImageNET e treinados por 10k iterações com taxa de aprendizado inicial $\alpha = 0,002$, que é reduzida a 10% do seu valor a partir da iteração 6k, e de mais 10% a partir da iteração 8k.

A Tabela I apresenta os resultados das detecções considerando somente as saídas dos 4 modelos (sem pós-processamento). O modelo que utiliza R50-C4 foi o que apresentou os valores mais baixos de AP_{50} e F_1 . Os modelos que utilizam a FPN (R50-FPN e R101-FPN) são os que apresentam maiores valores de AP_{50} . Estes mesmos modelos também são os que apresentam maiores valores de TP (91 e 89, respectivamente) mas, por outro lado, também são os que apresentam maiores valores de FP (43 e 45, respectivamente), o que acaba refletindo altos valores de revocação (Rc - 0,77 e 0,75, respectivamente) e em baixos valores de precisão (Pr - 0,68 e 0,66, respectivamente). O modelo que utiliza R101-C4 apresenta valores de TP (87) próximos aos dos modelos que possuem os maiores valores (91 e 89); em compensação, é o que apresenta o valor mais baixo de FN (25), resultando num bom compromisso entre precisão e Rc, e, conseqüentemente, no maior valor de F_1 .

B. Rastreamento dos objetos

Para as saídas de cada modelo, a etapa de pós-processamento descrita na Seção III-C foi aplicada, com $t = 5$ e $v \geq 50\%$, afim de agregar a informação temporal e espacial presente nos vídeos. Pode-se observar na Tabela I que esta etapa diminuiu substancialmente os valores de FP de todos os modelos, sacrificando poucos TP. Isto resultou em um aumento significativo em Pr e uma ligeira queda em Rc, refletindo em um aumento do F_1 mas uma baixa em AP_{50} .

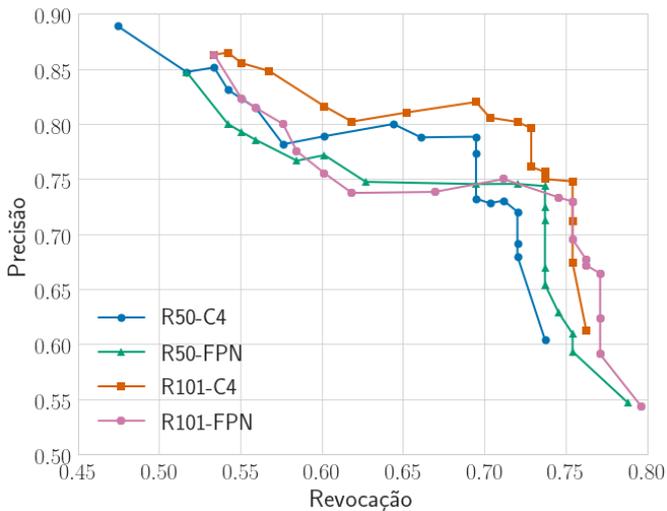
A Figura 4 apresenta a curva de *precisão* \times *revocação* para os resultados obtidos aplicando o pós-processamento utilizando $t = 33$ e variando v de 100% a 5%, com passo de 5%. Como pode ser observado, conforme diminui-se o valor de v , há uma queda em Pr e um ganho em Rc. O modelo R50-C4 alcança Pr próxima a 0,9 porém com baixa Rc. Os modelos R50-FPN e R101-FPN apresentam alta Rc mas baixa Pr, o que significa maiores quantidades de FP. Como já observado anteriormente, o modelo R101-C4 apresenta o melhor compromisso precisão e revocação, quando comparado aos demais modelos.

V. CONCLUSÃO

Nesse trabalho, apresentamos uma ampla base de dados de imagens aéreas contendo diversos objetos comumente associados aos focos de mosquito. Também propomos um sistema para detectar objetos de interesse, atuando em duas etapas. Na primeira etapa a detecção é feita quadro a quadro utilizando a arquitetura *Faster R-CNN*, e uma etapa subsequente de

TABELA I: Resultados detecção dos modelos, antes e após pós-processamento, para diferentes métricas de avaliação.

Backbone	Pós-processamento	AP ₅₀	TP	FP	FN	Pr	Rc	F ₁
R50-C4	-	65,10	81	33	37	0,71	0,67	0,70
	$t = 5, v \geq 50\%$	62,98	77	20	41	0,79	0,65	0,72
R50-FPN	-	72,85	91	43	27	0,68	0,77	0,72
	$t = 5, v \geq 50\%$	71,78	89	30	29	0,75	0,75	0,75
R101-C4	-	71,19	87	25	31	0,78	0,74	0,76
	$t = 5, v \geq 50\%$	69,79	84	20	34	0,81	0,71	0,76
R101-FPN	-	71,58	89	45	29	0,66	0,75	0,71
	$t = 5, v \geq 50\%$	70,45	87	29	31	0,75	0,74	0,74

Fig. 4: *Precisão × revocação* obtidos após pós-processamento utilizando $t = 33$ e variando $v = 0,05 : 0,05 : 1,0$.

rastreamento efetua o registro entre detecções sucessivas. Esse processamento adicional é capaz de reduzir os casos de falsos positivos e falsos negativos, melhorando significativamente o desempenho do sistema como um todo. Os resultados obtidos são promissores. Utilizando a ResNet-101-C4 como *backbone*, é possível obter uma acurácia de 0,76 em termos da *F1-score*.

AGRADECIMENTO

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001; FAPERJ; CNPq; e Google LARA 2019.

REFERÊNCIAS

- [1] C. Rückert, J. Weger-Lucarelli, S. M. Garcia-Luna *et al.*, “Impact of simultaneous exposure to arboviruses on infection and transmission by aedes aegypti mosquitoes,” *Nature Communications*, vol. 8, pp. 1–9, may 2017.
- [2] A. Laserna, J. Barahona-Correa, L. Baquero *et al.*, “Economic impact of dengue fever in latin america and the caribbean: A systematic review,” *Revista Panamericana de Salud Pública*, vol. 42, p. e111, sep 2018.
- [3] Pan American Health Organization, “Dengue in the Americas reaches highest number of cases recorded,” <https://reliefweb.int/report/world/dengue-americas-reaches-highest-number-cases-recorded>, 2019, accessed: 2020-03-22.
- [4] World Health Organization, “Zika virus,” <https://www.who.int/news-room/fact-sheets/detail/zika-virus>, 2018, accessed: 2020-03-22.
- [5] —, “Chikungunya,” <https://www.who.int/news-room/fact-sheets/detail/chikungunya>, 2017, accessed: 2020-03-22.
- [6] —, “Yellow fever – Brazil,” <https://www.who.int/csr/don/11-february-2019-yellow-fever-brazil/en/>, 2019, accessed: 2020-03-23.
- [7] L. Lambrechts and A. Failloux, “Vector biology prospects in dengue research,” *Memórias do Instituto Oswaldo Cruz*, vol. 107, no. 8, pp. 1080–1082, dec 2012.
- [8] W. L. Passos, T. M. Dias, H. M. A. Junior *et al.*, “Acerca da detecção automática de focos do mosquito aedes aegypti,” in *Brazilian Symposium on Telecommunications and Signal Processing*, Campina Grande, Brazil, sep 2018, pp. 392–396.
- [9] A. Agarwal, U. Chaudhuri, S. Chaudhuri *et al.*, “Detection of Potential Mosquito Breeding Sites Based on Community Sourced Geotagged Images,” in *Geospatial InfoFusion and Video Analytics IV; and Motion Imagery for ISR and Situational Awareness II*, no. 3, jul 2014, p. 90890M.
- [10] M. G. Prasad, A. Chakraborty, R. Chalasani *et al.*, “Quadcopter-based Stagnant Water Identification,” in *IEEE National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics*, Patna, India, dec 2015, pp. 1–4.
- [11] M. Mehra, A. Bagri, X. Jiang, and J. Ortiz, “Image Analysis for Identifying Mosquito Breeding Grounds,” in *IEEE International Conference on Sensing, Communication and Networking (Workshop)*, London, UK, jun 2016, pp. 1–6.
- [12] G. S. Andrade, T. M. Dias, V. C. Alves *et al.*, “Fighting back zika, chikungunya and dengue: Detection of mosquito-breeding habitats using an unmanned aerial vehicle,” *IEEE CAS Student Design Competition 2017–2018 Finalist project*, may 2018.
- [13] P. Haddawy, P. Wettayakorn, B. Nonthaleerak *et al.*, “Large scale detailed mapping of dengue vector breeding sites using street view images,” *PLOS Neglected Tropical Diseases*, vol. 13, no. 7, pp. 1–27, jul 2019.
- [14] T. M. Dias, V. C. Alves, H. M. Alves, L. F. Pinheiro, R. S. G. Pontes, G. M. Araújo, A. A. Lima, and T. M. Prego, “Autonomous detection of mosquito-breeding habitats using an unmanned aerial vehicle,” in *2018 Latin American Robotic Symposium, 2018 Brazilian Symposium on Robotics (SBR) and 2018 Workshop on Robotics in Education (WRE)*, 2018, pp. 351–356.
- [15] B. D. Barros, “On data augmentation techniques for the automatic detection of mosquito breeding grounds using videos,” *Dissertação de mestrado*, Universidade Federal do Rio de Janeiro - COPPE/PEE, Rio de Janeiro, RJ, Brasil, 2019.
- [16] W. L. Passos, “Automatic aedes aegypti breeding grounds detection using computer vision techniques,” *Dissertação de mestrado*, Universidade Federal do Rio de Janeiro - COPPE/PEE, Rio de Janeiro, RJ, Brasil, 2019.
- [17] Z. Zhang, “A Flexible New Technique for Camera Calibration,” *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 22, no. 11, pp. 1330–1334, nov 2000.
- [18] Z. Zhao, P. Zheng, S. Xu, and X. Wu, “Object Detection With Deep Learning: A Review,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 11, pp. 3212–3232, jan 2019.
- [19] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 39, no. 6, pp. 1137–1149, jun 2017.
- [20] B. S. Reddy and B. N. Chatterji, “An fft-based technique for translation, rotation, and scale-invariant image registration,” *IEEE Transactions on Image Processing*, vol. 5, no. 8, pp. 1266–1271, 1996.
- [21] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, USA, jun 2016, pp. 770–778.
- [22] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, “Feature Pyramid Networks for Object Detection,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, USA, jul 2017, pp. 936–944.